ÖREBRO UNIVERSITET
INSTITUTIONEN FÖR
NATURVETENSKAP OCH TEKNIK

# Inbyggda System, DT511G

## *Inbyggda System, teori*

## *A001*

## 4,5 högskolepoäng

## Skriftlig tentamen

**2023-03-20**

## Inbyggda system, DT511G (A001)

**Tillåtna hjälpmedel:** penna, radergummi, engelska-svenska ordbok

## Instruktioner:

***Läs igenom alla frågor noga.***
**Ange tentamenskoden på svarsdokumentet.**
**Du kan svara på *Svenska* eller *Engelska*.**
***Skriv tydligt* (gäller även för en digital tentamen).**
**Detta är en individuell examination - alla misstankar om otillåtet samarbete kommer att rapporteras.**
**Ansvarig lärare finns tillgänglig via telefon fr.o.m. andra skrivtimmen.**
**Skriv läsligt!**
**förklara och motivera era svar**

**Ansvarig lärare:** Pascal Rebreyend, tel: 0702001422

## För betyg G krävs 50% av total poäng (20 på 40)

**(26 poång gav betyg 4, och 32 poäng gav betyg 5)**

## *Lycka till!*

# Question 1 (4 points)

What are the differences between EDF and EDF* algorithm? How the EDF* algorithm works?

# Question 2 (6 points)

- We have a set of periodic tasks (see following table)

| Task | Period ( $p_i$ ) | Execution time ( $e_i$ ) |
|---|---|---|
| A | 25 | 5 |
| B | 11 | 2 |
| C | 15 | 2 |
| D | 7 | 2 |

  - Compute the Processor Utilization Factor. What is your conclusion? (2 points)
  - Using the Rate Monotonic Scheduling Algorithm, what is your conclusion? (2 points)
  - Calculate the maximum response times for all the task. (Priorities are set according to RMS) (2 points)

    **Notice:** The following equation is used for calculation of maximum response times:

$$R_i = e_i + \sum_{\tau_j \in H_i} \left\lceil \frac{R_i}{p_j} \right\rceil e_j$$

# Question 3 (4 points)

You have to implement a binary tree in an embedded system with an hard realtime operating system. (A binary tree is a tree for which each node has at the maximum two children). You know also that in your case the maximum height of the tree will be 5 and the value hold by each node is an integer between 0 and 321447. How you will implement this binary tree?

# Question 4 (3 points)

When in FreeRTOS we use a timer we associate with the time a function (called call-back) which will be called when the timer expires. Are they any constraints specific to these call-back functions and why?

# Question 5 (5 points)

We have a set of aperiodic tasks (see following table). All the tasks have the same arrival time and all are independent (they don't share any mutual ressource other than the single cpu of the system). Using the earliest due date (EDD) algorithm, is this set of tasks is schedulable? (2 points)

In case this algorithm is not able to find a feasible schedule, what will be your conclusion? Do you plan to try another algorithm? (3 points)

| Task | Deadline ( $d_i$ ) | Execution time ( $e_i$ ) |
|---|---|---|
| A | 5 | 1 |
| B | 6 | 3 |
| C | 3 | 2 |
| D | 8 | 2 |
| E | 2 | 1 |

# Question 6 (10 points)

You are writing the FreeRTOs software for a coffe and expresso machine.

This machine has the following sensors: (1 means high)

| Name of sensors | Pin connected to the sensor | Description |
|---|---|---|
| S_Water | P_ws | 1 if enough water in the tank, 0 otherwise |
| S_temp_ok | P_temp | 1 if temperature is hot enough<br>0 otherwise |
| S_slot_left | P_left | 1 if a cup is present in the left slot, 0 therwise |
| S_slot_right | P_right | 1 if a cup is present in the right slot, 0 therwise |
| S_B1 | P_B1 | 1 if the button 1 is pressed<br>0 otherwise |
| S_B2 | P_B2 | 1 if button 2 is pressed, 0 otherwise |

The machine has the following actuators. (They are on on when the corresponding pin is set to high)

| Name | Pin | Description |
|---|---|---|
| Heater | P_heat | Warm the water in the tank |
| Led_Red | P_red | Turn on the red led |
| Led_Green | P_Green | Turn on the green led |
| Coffee | P_Cafe | Prepare a coffee |
| Expresso | P_Expresso | Prepare an expresso |

You should write the FreeRTOS software which do the following and should lead to a fully functional coffee machine:

- If the user press the button 1, this means he wants an expresso delivered in the right slot
- If the user press the button 2 once, this means he wants a regular coffee delivered in the left slot
- If the user press the button 2 twice in less than 3 seconds, this means he wants a long coffee delivered in the left slot
- If the user asks for a drink in one slot and there is no cup in the slot, if the cup appear in less than 10 seconds and stay for more than 2 seconds, the beverage will be delivered. Otherwise, the order is canceled. During the 10 seconds, both the red and green light should flash until a cup appear.
- If the user asks for a drink and the water in the tank is not at the right temperature, the water should be heated until the temperature is ok and only then the beverage will be delivered
- If there is not enough water in the tank, no beverage will be delivered and the red light should be on
- When a beverage is order, if nothing is preventing its delivery, the green light should be on while the beverage is prepared. Once the beverage is delivered, the green light should flash until the cup is removed.
- To deliver an expresso, the pin P_expresso should be on high between 1250 and 1300ms
- To deliver a regular coffee, the pin P_coffee should be high between 300 to 350ms
- To deliver a long coffee, the pin P_coffee should be high between 500 to 520ms.

## Question 7 (8 points)

You are working in a company developping a real-time operating system. This real-time system has lot of functionalities, including mutex, binary semaphores, ... Unfortunatly, counting semaphores are not implemented. Your task is therefore to write a library which implement counting semaphores. Write how you will do it (idea and code)